

Aplikasi dan Optimasi Kombinatorial pada *Ant Colony*

Letivany Aldina / 13514067

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

13514067@std.stei.itb.ac.id

Abstract— Ants are insects of the tribe Formicidae, Hymenoptera nation which lives in colonies or form groups. One ant colonies consisting of thousands semut. An ant colony is also supported by the mastery of large areas to sustain their lives. These colonies then form a larger social entity called a superorganism. The existence of a social nature are high on this ant effect on life experienced ants. The social nature brings many positive effects such as the construction of the nest, the protection of the colony, foraging activity, and others. One of the interesting things of their activities is in the process of looking for food.

Keywords—Ant, Ant Colony Optimization (ACO), Combinatorial, Pheromone.

I. PENGENALAN SIFAT ALAMI SEMUT

Salah satu kebutuhan semut yang hidup berkelompok dalam koloni adalah komunikasi. Dari penemuan dan percobaan yang telah dilakukan, diketahui bahwa diantara semut-semut tersebut terdapat zat *pheromone* yang membantu semut dalam berkomunikasi, terutama saat terjadi serangan yang dilakukan oleh musuh dan transportasi dalam mencari makanan.

Proses terjadinya komunikasi ini dimulai dengan semut pertama sebagai pemandu semut-semut di belakangnya dalam berjalan. Semut pertama ini meninggalkan zat *pheromone* dalam jumlah tertentu sebagai penanda jalan. Hal ini menyebabkan semut-semut berikutnya dapat mengidentifikasi zat *pheromone* tersebut dan mengetahui bahwa jalan inilah jalan yang digunakan oleh semut sebelumnya.

Zat *pheromone* ini sewaktu-waktu dapat menguap. Uniknya, tidak hanya semut pertama yang meninggalkan jejak *pheromone* tersebut, namun juga diikuti dengan semut-semut berikutnya. Hal ini dapat menjamin semut-semut lain yang berada di urutan paling belakang tidak akan kehilangan arah dan tersesat. Semakin kuat jejak *pheromone* yang ditinggalkan, maka dapat diketahui jumlah semut yang melewati jalan tersebut juga banyak.

Percobaan mengenai perilaku semut pertama kali dilakukan oleh seorang peneliti Biologi, Richard Feynman. Richard Feynman meletakkan sebongkah gula di salah satu ujung bak mandi, lalu menunggu seekor semut datang dan menemukannya. Ketika semut yang pertama kali datang kemudian kembali ke sarangnya, Feynman mengamati jejaknya yang berkelok.

Kemudian Feynman mengikuti jejak-jejak semut yang datang berikutnya. Feynman menemukan bahwa semut yang datang belakangan tidak mengikuti jejak yang ditinggalkan, mereka lebih pintar, mereka mengambil jalan memotong sampai akhirnya jejaknya mendekati garis lurus.

Berdasarkan penelitian yang dilakukan Feynman, seorang ahli komputer bernama Alfred Bruckstein membuktikan secara matematis bahwa semut-semut yang datang selanjutnya memang meluruskan jejak yang berkelok-kelok tersebut.

II. ANT COLONY OPTIMIZATION

Ant Colony Optimization merupakan teknik pencarian *multi-agent* untuk menyelesaikan permasalahan optimasi kombinatorial yang diinspirasi tingkah laku semut dalam suatu koloni (Marco, 2004). Algoritma *Ant Colony* pertama kali diperkenalkan oleh Marco Dorigo pada tahun 1992 sebagai thesis PhD-nya yang kemudian dipublikasikan dengan nama *Ant System (AS)*.

ACO merupakan sebuah probabilitas komputasi teknik untuk memecahkan masalah yang dapat dikurangi untuk menemukan jalur yang baik melalui grafik. ACO meniru perilaku koloni semut dalam mencari sumber makanan bermula dari dan kembali ke sarangnya yang secara alami mencari rute terpendek.

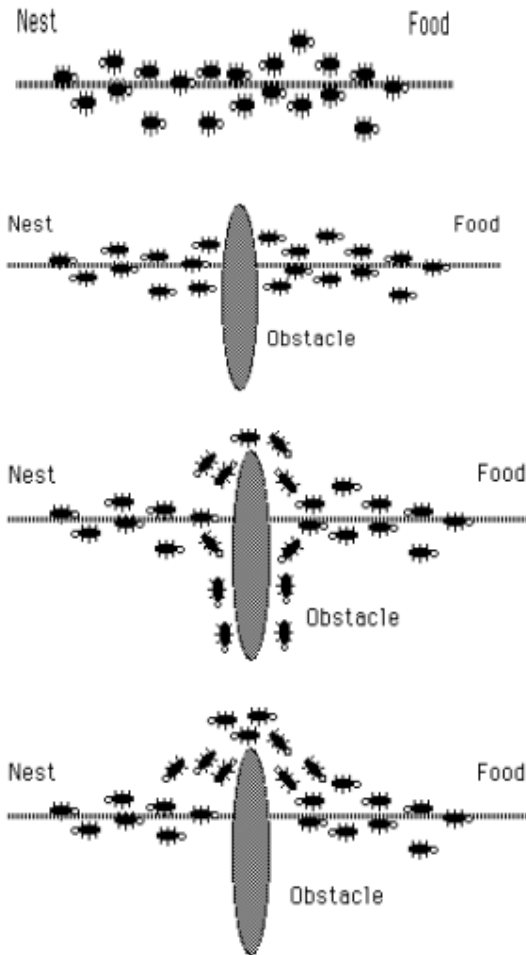
ACO termasuk dalam kelompok *Swarm Intelligence*, yaitu salah satu jenis pengembangan paradigma yang digunakan untuk menyelesaikan masalah optimasi, dimana inspirasi yang digunakan untuk memecahkan masalah tersebut berasal dari perilaku kumpulan atau kawanan.

ACO biasanya digunakan untuk menyelesaikan *discrete optimization* dan persoalan yang kompleks dimana terdapat banyak variabel. Hasil yang diperoleh dengan menggunakan ACO, walaupun tidak optimal tetapi mendekati optimal. ACO sudah diterapkan di berbagai masalah seperti VRP, penjadwalan proyek dengan sumber daya terbatas, data mining, penjadwalan pekerjaan (*job scheduling*), dan beberapa masalah kombinatorial yang lain.

Dari proses komunikasi dalam mencari makanan yang telah dijabarkan sebelumnya, hal menarik dari perilaku semut-semut tersebut adalah kemampuannya dalam

menemukan jarak terpendek antara sarang mereka dan sumber makanan.

Dari keadaan tersebut, semut yang melewati lintasan yang pendek akan meninggalkan jejak *pheromone* yang lebih tajam daripada semut yang melewati lintasan yang lebih panjang. Juga menjadi sifat alami semut untuk mengikuti jalan yang memiliki jejak *pheromone* yang lebih kuat.



Pada saat sistem (lingkungan semut) telah menemukan solusi yang optimal, yaitu lintasan terpendek, maka ACO akan dapat beradaptasi dengan cepat terhadap perubahan yang terjadi di sekitar. Adaptasi ini didasarkan pada *pheromone* yang merupakan dasar dari *ant-system*. *Pheromone* yang lebih kuat akan dimiliki oleh solusi dengan jalur yang lebih optimal pada akhir suatu algoritma.

ACO dalam prosesnya dimulai dengan semut secara individual memiliki kemampuan kognitif terbatas secara kolektif mampu menemukan jalur terpendek antara sumber makanan dan sarang. Misalkan terdapat N semut dalam satu koloni. Semut-semut memulai perjalanan dari sarang menuju sumber makanan dengan melalui beberapa titik dan berakhir pada titik akhir setiap siklus atau iterasi. Jika semua semut sudah menyelesaikan perjalanannya,

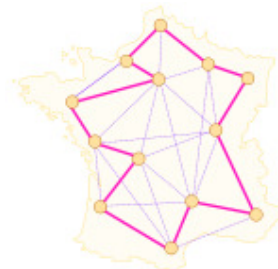
jumlah *pheromone* pada lintasan terbaik secara global akan diperbaharui. Lintasan global terbaik artinya terbaik diantara semua semut. Pada awal proses, semua ruas dari titik awal akan diberi jumlah *pheromone* yang sama. Pada iterasi pertama, semua semut akan mulai dari titik awal dan berakhir pada titik terakhir dengan memilih titik-titik secara acak. Proses optimasi berakhir jika jumlah iterasi maksimum sudah tercapai atau tidak ada lagi solusi yang lebih baik yang bisa didapatkan dalam beberapa iterasi yang berurutan.



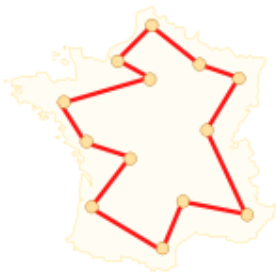
1



2



3



4

III. FUNGSI

A. Prilaku Semut

Semut bergerak dari titik i ke titik j dengan probabilitas:

$$p_{i,j} = \begin{cases} \frac{\tau_{i,j}^\alpha}{\sum_{j \in N_i^{(k)}} \tau_{i,j}^\alpha}, & \text{jika } j \in N_i^{(k)} \\ 0, & \text{jika } j \notin N_i^{(k)} \end{cases}$$

Dalam persamaan tersebut α menunjukkan derajat kepentingan *pheromone* dan $N_i^{(k)}$ adalah pilihan yang dipunyai semut k (*neighborhood*) pada saat ia berada pada titik i . *Neighborhood* dari semut k pada titik i akan mengandung semua titik yang bisa dituju yang tersambung langsung ke titik i , kecuali titik yang sudah dikunjungi sebelumnya.

B. Penambahan dan Penguapan Pheromone

Seekor semut k ketika melewati ruas akan meninggalkan *pheromone*. Jumlah *pheromone* yang terdapat pada ruas ij setelah dilewati semut k diberikan dengan rumus:

$$\tau_{i,j} \leftarrow \tau_{i,j} + \Delta\tau^k$$

Dengan meningkatnya nilai *pheromone* pada ruas i - j , maka kemungkinan ruas ini akan dipilih lagi pada iterasi berikutnya semakin besar. Setelah sejumlah titik dilewati maka akan terjadi penguapan *pheromone* dengan aturan sebagai berikut:

$$\tau_{i,j} \leftarrow (1 - \rho)\tau_{i,j}; \forall (i,j) \in A$$

Dimana $\rho \in (0,1]$ adalah parameter tingkat penguapan dan A menyatakan segmen atau ruas yang sudah dilalui oleh semut k sebagai bagian dari lintasan dari sarangnya menuju ke sumber makanan. Penurunan jumlah *pheromone* memungkinkan semut untuk mengeksplorasi lintasan yang berbeda selama proses pencarian. Ini juga akan menghilangkan kemungkinan memilih lintasan yang kurang bagus. Selain itu, ini juga membantu membatasi nilai maksimum yang dicapai oleh suatu lintasan

pheromone. Jumlah *pheromone* yang ditambahkan pada ruas i - j oleh semut k diberikan sebagai berikut:

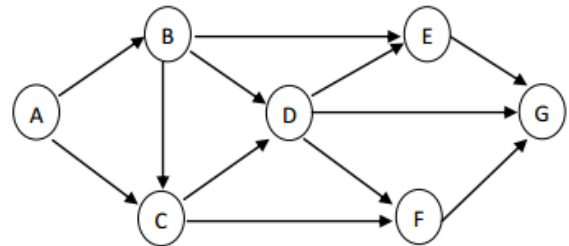
$$\Delta\tau_{i,j}^{(k)} = \frac{Q}{L_k}$$

Dimana Q adalah konstanta dan L_k adalah lintasan terpendek yang dilalui semut k . Nilai Q biasanya ditentukan oleh user. Atau bisa juga diimplementasikan dengan cara berikut:

$$\Delta\tau_{i,j}^{(k)} \begin{cases} \frac{Q}{L_k}, & \text{jika } (i,j) \in \text{lintasan terbaik global} \\ 0, & \text{untuk yang lain} \end{cases}$$

IV. PERMASALAHAN LINTASAN TERPENDEK

Lintasan terpendek yang harus dilalui oleh semut-semut juga mempunyai masalah dalam pemecahannya. Namun, hal ini dapat dipecahkan dengan menggunakan teori graf. Misalkan gambar di bawah merupakan graf ABCDEFG yang berarah dan tidak berbobot.



Misalkan titik awal adalah titik A, dan titik akhir adalah titik G. Agar bisa sampai di titik G, ada beberapa jalur yang dapat digunakan:

- A -> B -> C -> D -> E -> G
- A -> B -> C -> D -> F -> G
- A -> B -> C -> D -> G
- A -> B -> C -> F -> G
- A -> B -> D -> E -> G
- A -> B -> D -> F -> G
- A -> B -> D -> G
- A -> B -> F -> G
- A -> C -> D -> E -> G
- A -> C -> D -> F -> G
- A -> C -> D -> G
- A -> C -> F -> G

Berdasarkan contoh jalur di atas, dapat dihitung jalur terpendek dalam mencari jarak antara dua titik.

IV. PENYELESAIAN MASALAH OPTIMASI

Penyelesaian masalah pencarian jalur terpendek dalam penerapan ACO dapat dilakukan dengan dua metode, yaitu metode konvensional dan metode heuristik. Metode konvensional diterapkan dengan

perhitungan matematis biasa, sedangkan metode heuristik diterapkan dengan perhitungan kecerdasan buatan, dengan menentukan basis pengetahuan dan perhitungan.

1. Metode Konvensional
Metode konvensional berupa algoritma yang menggunakan perhitungan matematis biasa. Ada beberapa metode yang biasa digunakan untuk melakukan pencarian jalur terpendek, diantaranya algoritma Djikstra, algoritma Floyd – Warshall, dan algoritma Bellman – Ford.
2. Metode Heuristik
Metode heuristik adalah sub-bidang dari kecerdasan buatan yang digunakan untuk melakukan pencarian dan optimasi. Ada beberapa algoritma pada metode heuristik yang biasa digunakan dalam permasalahan optimasi, diantaranya algoritma genetika, algoritma ACO merupakan salah satunya, logika fuzzy, jaringan syaraf tiruan, pencarian tabu, *simulated annealing*, dan lain-lain.

IV. VARIASI POPULER ALGORITMA ACO

- A. *Elitis Sistem Ant*
Solusi terbaik global deposito pheromone pada setiap iterasi bersama dengan semua semut lain.
- B. *Max-Min Ant System (MMAS)*
 1. Ditambahkan maksimum dan minimum jumlah pheromone
 2. Hanya global iterasi terbaik atau disetor wisata terbaik pheromone
 3. Semua tepi-tepi yang telah siap untuk melakukan r max dan diinisialisasi kembali ketika mendekati stagnasi.
- C. Pseudo-acak proporsional aturan
- D. *Rank* berbasis *Ant System* (Asrank)
- E. Pemecahan kesulitan dalam definisi
Dengan algoritma ACO, jalan terpendek dalam grafik, misalnya antara dua titik A dan B, dapat dibangun dari kombinasi beberapa jalan. Tidak mudah untuk memberikan definisi yang tepat mengenai algoritma *ant colony*. Secara umum, algoritma *ant colony* dianggap sebagai solusi *metaheuristics* yang masing-masing diwakili oleh semut yang bergerak dalam ruang pencarian. Semut menandai solusi terbaik dan memperhatikan jejak-jejak yang sebelumnya ditinggalkan untuk mengoptimalkan pencarian mereka.

VI. METODE LAIN YANG TERKAIT ACO

- A. Algoritma Genetik (GA), yaitu algoritma yang lebih mempertahankan untuk menggunakan banyak solusi. Proses pencarian solusi menggunakan prinsip evolusi. Solusi dengan kualitas sedang

digabungkan untuk mendapatkan solusi baru yang lebih baik, sedangkan solusi berkualitas rendah dibuang.

- B. *Simulated Annealing* (SA), teknik pengoptimalan yang mencari solusi melintasi ruang pencarian dengan menggunakan parameter-parameter tertentu.
- C. *Tabu Search* (TS), mirip dengan *Simulated Annealing*, namun menghasilkan banyak solusi yang dimutasi bergerak ke arah solusi yang mempunyai kesesuaian terendah sehingga mencegah gerakan solusi yang lebih besar.
- D. Sistem Kekebalan Buatan (AIS), yaitu algoritma yang meniru sistem kekebalan tubuh vertebrata.

V. SEJARAH SINGKAT

- 1959, Pierre-Paul Grass menemukan teori Stigmergy untuk menjelaskan perilaku bangunan sarang rayap;
- 1983, Deneubourg dan rekan-rekannya mempelajari perilaku kolektif dari semut;
- 1988, Moysen Manderick memiliki artikel tentang organisasi diri diantara semut;
- 1989, Goss, Aron, dan Pasteels di Deneubourg memberikan ide algoritma optimisasi koloni semut;
- 1989, pelaksanaan model perilaku untuk makanan oleh Ebling dan rekan-rekannya;
- 1991, M.Dorigo mengusulkan *Ant System* bersama V. Maniezzo dan A. Colorni;
- 1995, Bilchev mempublikasikan *Parmee* yaitu usaha pertama untuk beradaptasi dengan masalah-masalah yang berkelanjutan;
- 1996, penerbitan artikel di *Ant*;
- 1996, Hoss dan Stutzle menciptakan *MAX-MIN Ant System*;
- 1997, Dorigo dan Gambardella menerbitkan *Ant Colony*;
- 1997, Martinoli dan rekan-rekannya memakai algoritma ACO untuk mengendalikan robot;
- 1998, Dorigo meluncurkan konferensi pertama yang didedikasikan untuk algoritma ACO;
- 1998, Stutzle mengusulkan awal implementasi paralel;
- 1999, Bonabeau dan koleganya telah menerbitkan sebuah buku tentang semut;
- 1999, aplikasi pertama untuk kendaraan *routing*, kuadrat penugasan, serta masalah multi-dimensi;
- 2000, penerbitan jurnal khusus tentang algoritma ACO;
- 2000, aplikasi pertama mengenai penjadwalan, penjadwalan urutan, dan kepuasan kendala;
- 2000, Gutjahr memberikan bukti pertama konvergensi untuk algoritma *ant colony*;
- 2001, penggunaan algoritma ACO pertama oleh perusahaan Eurobios dan AntOptima;
- 2001, Ireda dan rekan-rekannya menciptakan algoritma multi-objektif pertama;
- 2002, aplikasi pertama dalam perancangan jadwal dan jaringan Bayesian;

2002, Bianchi dan rekan-rekannya mengusulkan algoritma stokastik masalah pertama;

2004, Zlochin dan Dorigo menciptakan algoritma yang setara dengan gradien stokastik keturunan, menggunakan lintas-entropi dan algoritma perkiraan distribusi;

2005, aplikasi lipat protein yang pertama.

VI. CONCLUSION

Inspirasi yang diperlihatkan oleh kegiatan semut dalam mencari makanan pada dasarnya mempunyai perhitungan dasat matematis di dalamnya yakni kombinatorial dan graf. Dari perjalanan semut-semut tersebut, dapat diketahui bahwa semut mempunyai kemampuan kognitif yang dapat mengoptimasi perjalanannya dalam mencari makanan. Inspirasi dari kemampuan kognitif ini menghasilkan banyak penerapan-penerapan terutama dalam pembuatan algoritma optimasi.

REFERENCES

- [1] <http://www.metode-algoritma.com/2013/06/optimasi-kombinatorial-ant-colony.html>
Diakses pada tanggal 10 Desember 2015, pukul 13.00 WIB
- [2] <https://bsantosa.files.wordpress.com/2012/05/ant-colony-optimization1.pdf>
Diakses pada tanggal 10 Desember 2015, pukul 13.00 WIB

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 8 Desember 2015

Ttd



Letivany Aldina
13514067